What are the roles of the HTML audio and video element, and how do they work?

The audio and video elements allow you to add sound and video content to your HTML documents. The audio element supports popular audio formats like mp3, wav, and ogg. The video element supports mp4, ogg, and webm formats.

To include audio content on your web page, you can use the audio element with the src attribute pointing to the location of your audio file:

<audio src="CrystalizeThatInnerChild.mp3"></audio>

If you try to run this example, you'll see that nothing shows up on the page. However, if you inspect the page with the developer tools, you'll see the audio element is indeed on the page. If you want to see the audio player on the page, then you can add the audio element with the controls attribute:

<audio src="CrystalizeThatInnerChild.mp3" controls></audio>

The controls attribute enables users to manage audio playback, including adjusting volume, and pausing, or resuming playback. The controls attribute is a boolean attribute that can be added to an element to enable built-in playback controls. If omitted, no controls will be shown.

Apart from the src and controls attributes, there are several other attributes that enhance the functionality of the audio element. The loop attribute is a boolean attribute that makes the audio replay continuously. Here's an example of using the loop attribute to play one of Quincy Larson's songs titled "Can't stay down":

<audio

src="https://cdn.freecodecamp.org/curriculum/js-music-player/can't-stay-down.mp3"

loop

controls

></audio>

When the song reaches the end, it will loop back around and play it again from the beginning. Another attribute you can use is the muted attribute. When present in the audio element, this boolean attribute will start the audio in a muted state. Here's an example of using the muted attribute:

<audio

src="https://cdn.freecodecamp.org/curriculum/js-music-player/can't-stay-down.mp3"

loop

controls

muted

></audio>

When you start the song in the browser, you'll not hear anything. To hear the music you will need to click on the volume icon.

When it comes to audio file types, there are differences in which browsers support which type. To accommodate this, you can use source elements inside the audio element and the browser will select the first source that it understands. Here's an example of using multiple source elements for an audio element:

<audio controls>

<source src="audio.ogg" type="audio/ogg" />

<source src="audio.wav" type="audio/wav" />

<source src="audio.mp3" type="audio/mpeg" />

</audio>

The browser will first start with the ogg type, and if it can't play the audio, then it'll move down to the next type in the list.

All the attributes we have learned so far are also supported in the video element. Here's an example of using a video element with the loop, controls, and muted attributes:

<video

src="https://archive.org/download/BigBuckBunny\_124/Content/big\_buck\_bunny\_720p\_surround.mp4"

loop

controls

muted

></video>

For the src, or source attribute, we are using a video called "Big Buck Bunny" from archive.org. If you wanted to display an image while the video is downloading, you can use the poster attribute. This attribute is not available for audio elements and is unique to the video element. Here's an example of using the poster attribute with content provided by peach.blender.org:

<video

src="https://archive.org/download/BigBuckBunny\_124/Content/big\_buck\_bunny\_720p\_surround.mp4"

loop

controls

muted

poster="https://peach.blender.org/wp-content/uploads/title\_anouncement.jpg?x11217"

width="620"

></video>

This example is also using the width attribute to set the width to 620 pixels so the video fits better on the screen.